

### AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions of claims in the application:

#### **Listing of Claims:**

1. (Currently Amended) A software development system embodied on a computer-readable storage medium, comprising:
  - a shelving component that captures a current state of an intermediate software design containing pending changes developed on a private workspace, the shelving component removes the pending changes from the software design on the private workspace after the current state has been captured; and
  - a version control component that processes the intermediate software design as [[if]] a completed software design.
2. (Original) The system of claim 1, further comprising one or more code development systems that are employed by a developer to design software applications.
3. (Original) The system of claim 1, the shelving component is executed on at least one of a local development system or a centralized server or servers.
4. (Original) The system of claim 1, the version control component includes committed work or files that are generally checked in as finalized versions of code.
5. (Original) The system of claim 1, the version control system enables authorized users to retrieve respective versions of code to produce intermediate versions or builds of software in cases that utilize the shelving component or final versions or builds of software in cases that build from the committed work.

6. (Original) The system of claim 1, the shelving component is employed for at least one of an interrupted workflow application, a checkpoint application, a shared work application, a code backup application, a work exchange application, and a private workspace exchange application.
7. (Currently Amended) The system of claim 1, further comprising an unshelving component that restores ~~[[a]]~~ the private workspace to a state that was previously archived.
8. (Currently Amended) The system of claim 7, the unshelving component allows removing the state that is stored on a server or ~~to preserve~~ preserving changes shelved on the server in order that the changes are available for unshelving by other users.
9. (Currently Amended) The system of claim 1, further comprising a component to store state information for all files or folders in ~~[[a]]~~ the private workspace or for individual files in the private workspace.
10. (Currently Amended) The system of claim 9, the state information for all files and folders includes at least one of a unique identifier for all files and folders in the workspace and a version number of each file or folder in the workspace.
11. (Currently Amended) The system of claim 9, the state information for individual files includes at least one of a unique identifier for an individual file, a number of a version that was modified to create a current state, and a locally modified state of a file.
12. (Currently Amended) The system of claim 9, further comprising storing meta-data that is associated with the state information.

13. (Currently Amended) ~~A computer readable medium having computer readable instructions stored thereon for implementing the shelving component and the version control component of claim 1~~ The system of claim 7, the unshelving component merges unshelved changes with changes pending on the private workspace when an unshelve operation is initiated.
14. (Currently Amended) A code development system embodied on a computer-readable storage medium, comprising:  
    means for developing pending changes on non-finalized software on a private workspace;  
    means for archiving the non-finalized software with pending changes ~~in~~ to a version control system;  
    means for capturing one or more states associated with the non-finalized software; ~~and~~  
    means for removing the pending changes from the non-finalized software on the private workspace; and  
    means for processing the archived non-finalized software along with finalized software on the version control system.
15. (Original) The system of claim 14, further comprising means for restoring the states and the non-finalized software to a previous state.
16. (Currently Amended) A method to facilitate code development, comprising:  
    creating a version of software in a ~~local~~ private development system;  
    automatically determining at least one state for the software; ~~and~~  
    shelving the software and the state on a version control system[[]]; and  
    unshelving the version of software to a private workspace in accordance with the state,  
the version of software is deleted from the version control system when unshelving is initiated by an owner of the software, and the version of software is preserved on the version control system when unshelving is initiated by a non-owner of the software.

17. (Currently Amended) The method of claim 16, ~~further comprising unshelving the version of software in accordance with the state~~ unshelving the version of software to a private workspace that contains other pending changes causes the pending changes to be merged with the unshelved version of software.

18. (Original) The method of claim 16, further comprising providing at least one of a shelving command and an unshelving command to facilitate development of the software.

19. (Original) The method of claim 18, the shelving command is associated with at least one of a preserve option, a workspace option, a replace option, a comment option, a file option, an error condition, and an exit code.

20. (Original) The method of claim 18, the shelving and the unshelving command are associated with at least one security parameter.

21. (Original) The method of claim 18, the unshelving command is associated with at least one of a preserve option, a file option, a name option, a username option, an error condition, and an exit code.

22. (Currently Amended) A user interface embodied on a computer-readable storage medium for archiving intermediate versions of code, comprising:

a display component to highlight one or more ~~intermediate~~ files on a private workspace to archive on a version control system, the files contain pending changes associated with an intermediate version of code; and

a[[n]] shelving command input to enable users to ~~transfer~~ archive the intermediate files with pending changes to a version control system as if the intermediate files were finalized versions of the intermediate files, archiving removes the pending changes from the intermediate files on the private workspace.

23. (Currently Amended) The user interface of claim 21, further comprising an unshelving command that restores local versions of selected files on the version control system to a previous state, the selected files are deleted from the version control system when the unshelving command is initiated by an owner of the files, and the selected files are maintained on the version control system when the unshelving command is initiated by a non-owner of the files.